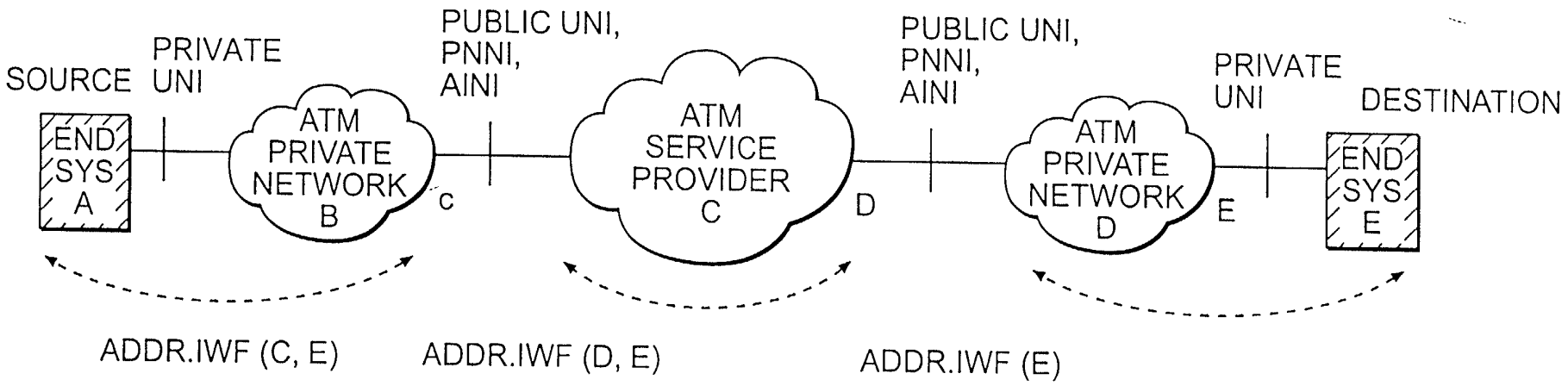


----------►	ROUTING DOMAIN
ADDR.IWF (X, Y)	ADDRESSING INTERWORKING FUNCTION USING ADDRESS PAIR WHERE 'X' IS THE ROUTING ADDRESS OF THE LOCAL DOMAIN AND 'Y' IS THE ADDRESS OF THE DESTINATION END SYSTEM NOTE: AT THE LAST DOMAIN, 'X' SHOULD BE EQUAL TO 'Y', HENCE, NO INTERWORKING FUNCTION IS REQUIRED.

FIG. 1

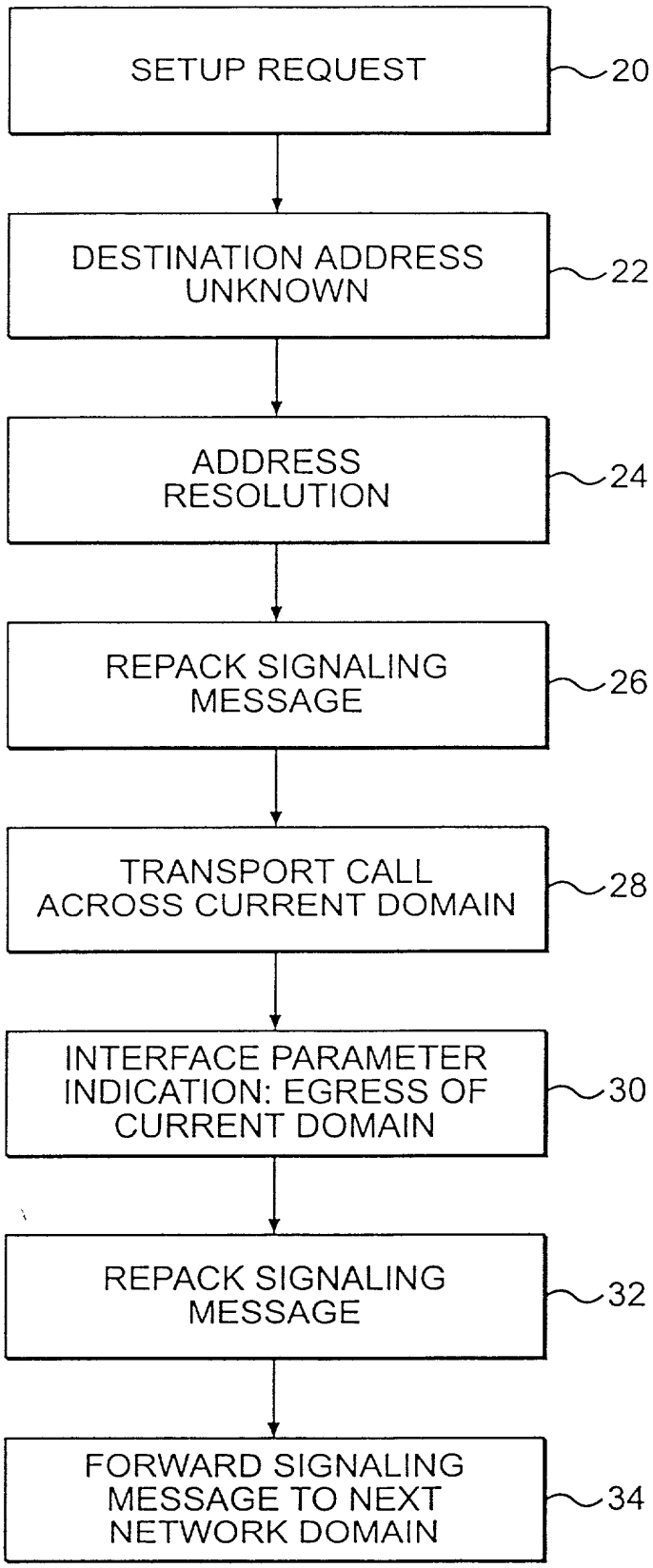

SETUP REQUEST
20
DESTINATION ADDRESS UNKNOWN
22
ADDRESS RESOLUTION
24
REPACK SIGNALING MESSAGE
26
TRANSPORT CALL ACROSS CURRENT DOMAIN
28
INTERFACE PARAMETER INDICATION: EGRESS OF CURRENT DOMAIN
30
REPACK SIGNALING MESSAGE
32
FORWARD SIGNALING MESSAGE TO NEXT NETWORK DOMAIN
34

FIG. 2

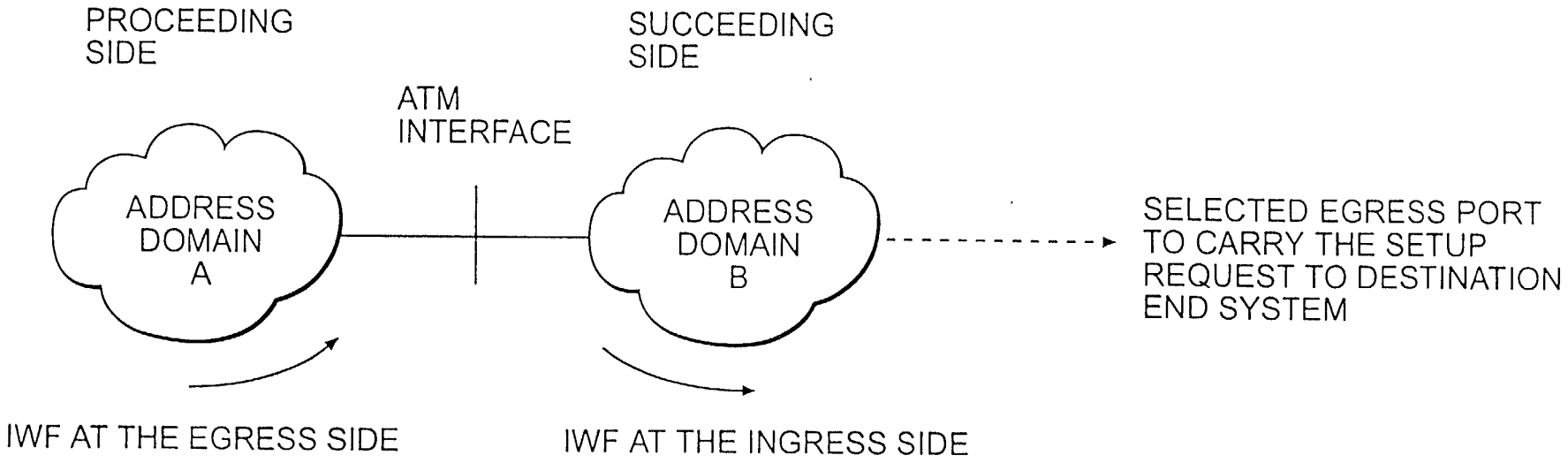

PROCEEDING SIDE
SUCCEEDING SIDE
ATM INTERFACE
ADDRESS DOMAIN A
ADDRESS DOMAIN B
SELECTED EGRESS PORT TO CARRY THE SETUP REQUEST TO DESTINATION END SYSTEM
IWF AT THE EGRESS SIDE
IWF AT THE INGRESS SIDE

FIG. 3

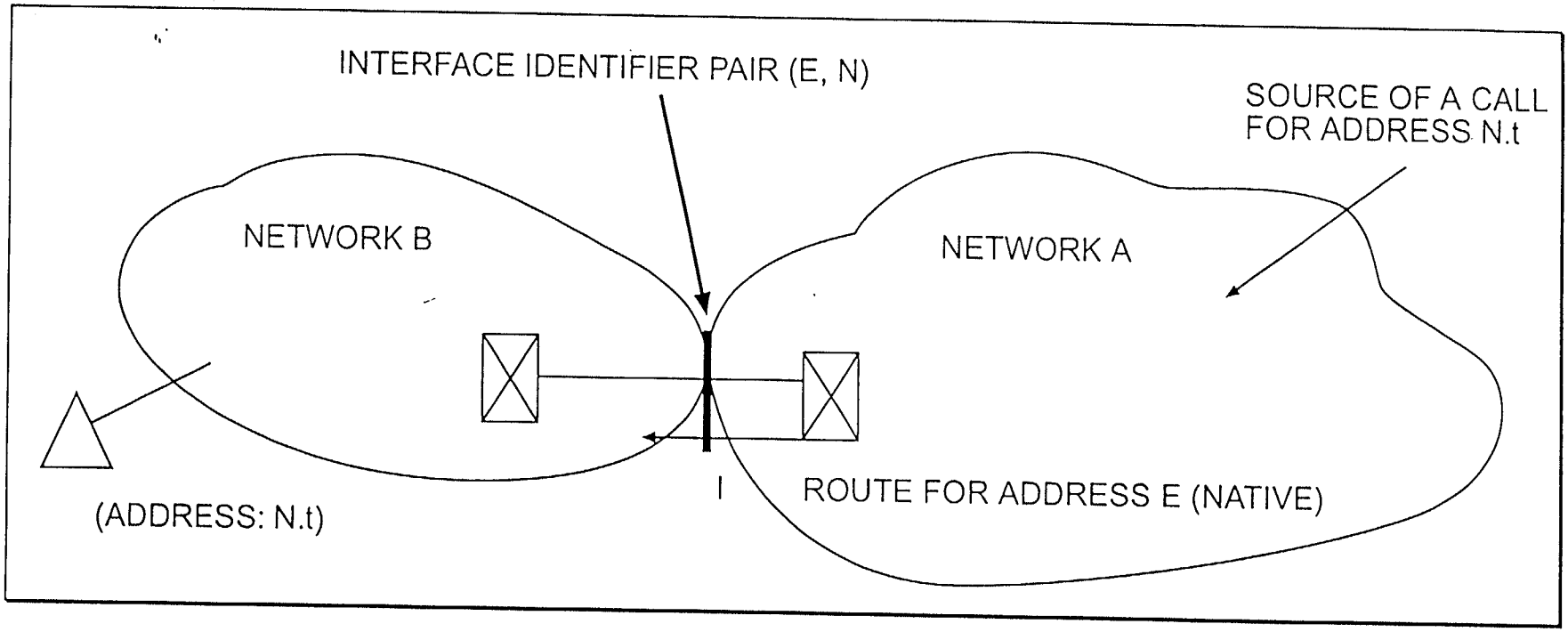

INTERFACE IDENTIFIER PAIR (E, N)
SOURCE OF A CALL FOR ADDRESS N.t
NETWORK B
NETWORK A
(ADDRESS: N.t)
ROUTE FOR ADDRESS E (NATIVE)

FIG. 4

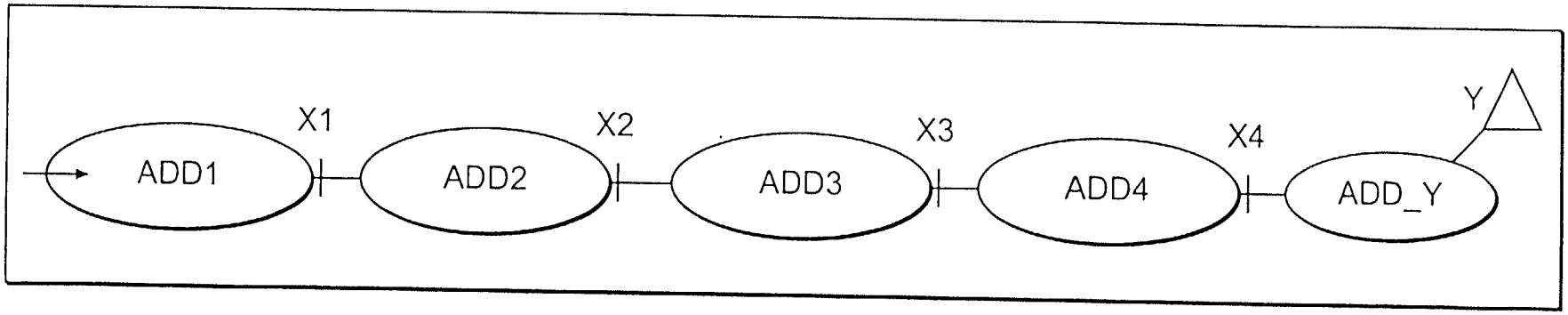

X1
X2
X3
X4
Y
ADD1
ADD2
ADD3
ADD4
ADD_Y

FIG. 5

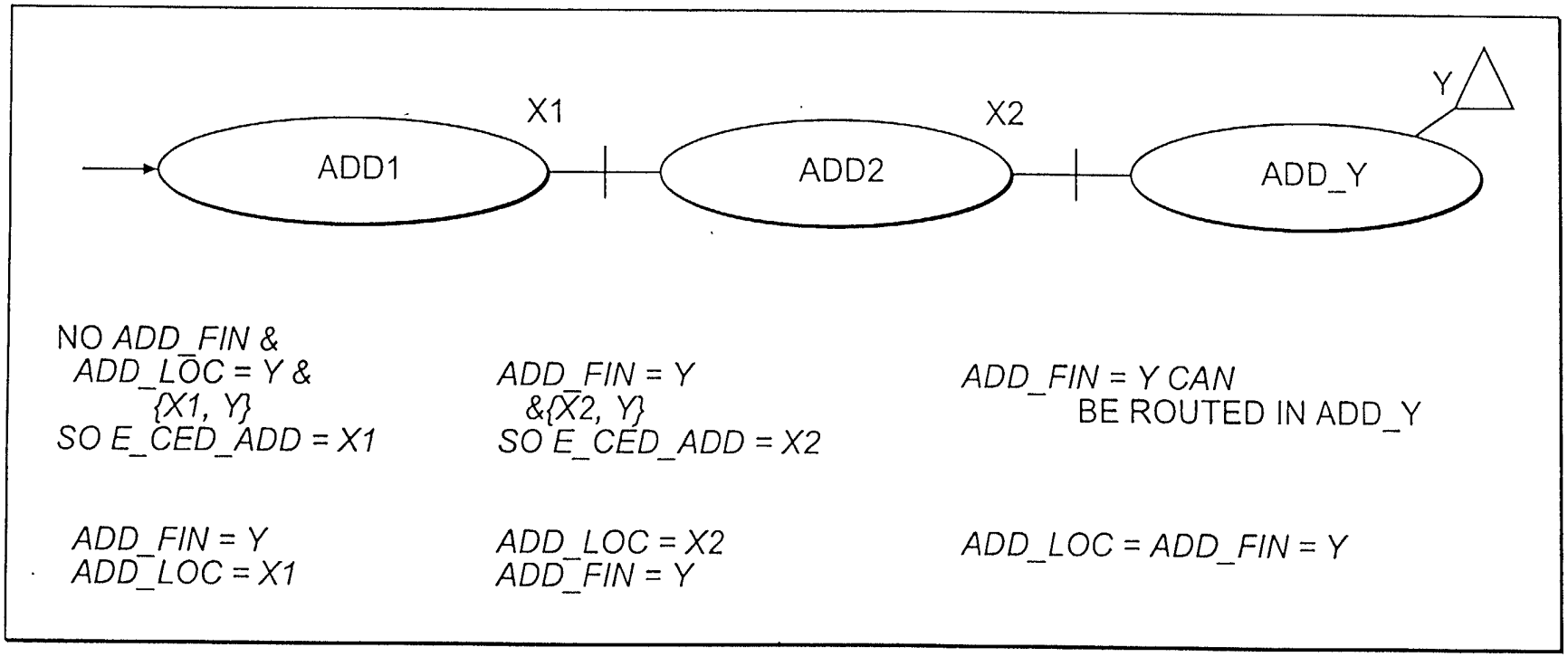

X1
X2
Y
ADD1
ADD2
ADD_Y
NO ADD_FIN &
ADD_LOC = Y &
{X1, Y}
SO E_CED_ADD = X1
ADD_FIN = Y
&{X2, Y}
SO E_CED_ADD = X2
ADD_FIN = Y CAN
BE ROUTED IN ADD_Y
ADD_FIN = Y
ADD_LOC = X1
ADD_LOC = X2
ADD_FIN = Y
ADD_LOC = ADD_FIN = Y

FIG. 6

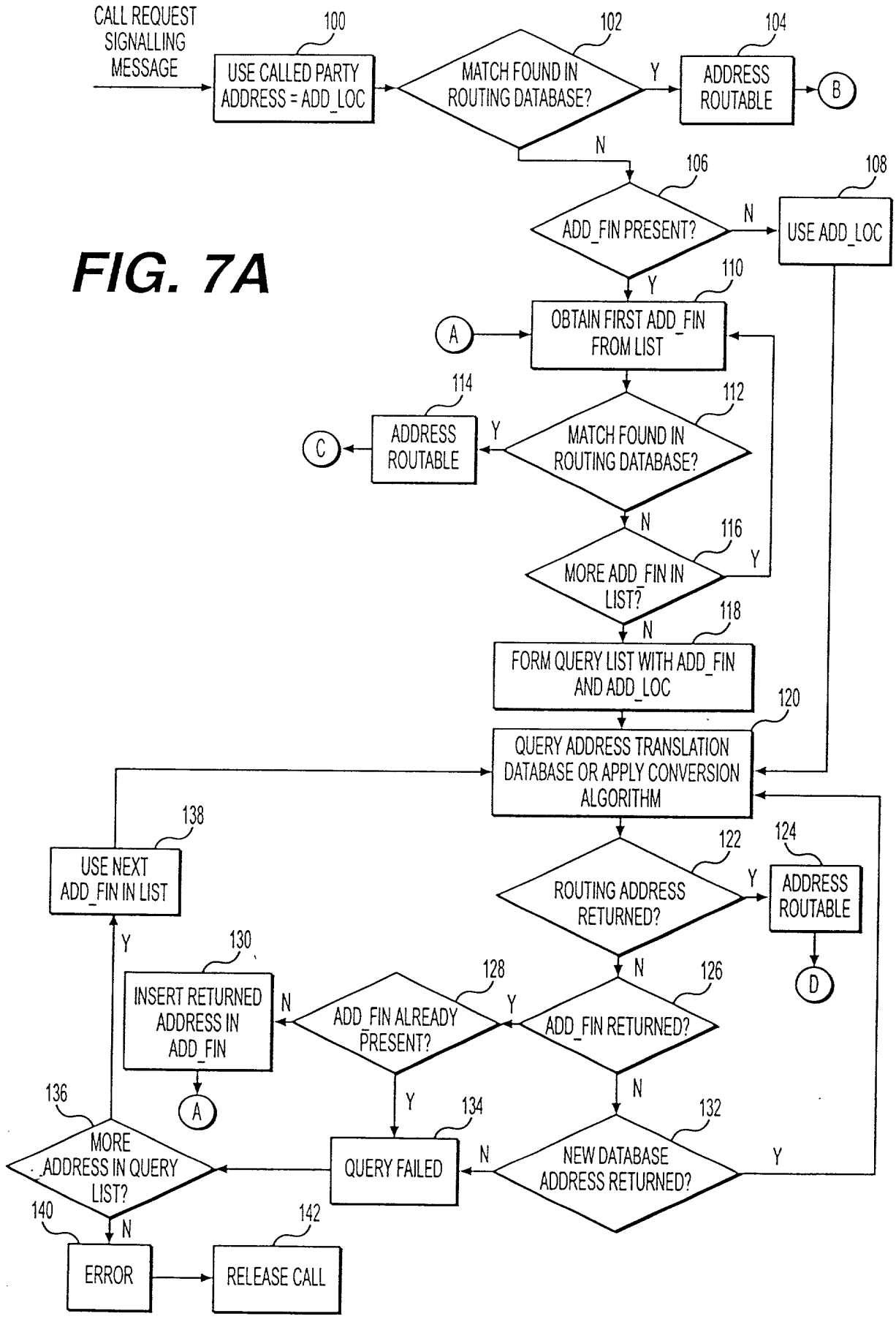

CALL REQUEST SIGNALLING MESSAGE
100
USE CALLED PARTY ADDRESS = ADD_LOC
102
MATCH FOUND IN ROUTING DATABASE?
Y
104
ADDRESS ROUTABLE
B
N
106
ADD_FIN PRESENT?
N
108
USE ADD_LOC
Y
110
OBTAIN FIRST ADD_FIN FROM LIST
A
114
ADDRESS ROUTABLE
C
Y
112
MATCH FOUND IN ROUTING DATABASE?
N
116
MORE ADD_FIN IN LIST?
Y
N
118
FORM QUERY LIST WITH ADD_FIN AND ADD_LOC
120
QUERY ADDRESS TRANSLATION DATABASE OR APPLY CONVERSION ALGORITHM
122
ROUTING ADDRESS RETURNED?
Y
124
ADDRESS ROUTABLE
D
N
126
ADD_FIN RETURNED?
Y
128
ADD_FIN ALREADY PRESENT?
N
130
INSERT RETURNED ADDRESS IN ADD_FIN
A
N
132
NEW DATABASE ADDRESS RETURNED?
Y
Y
134
QUERY FAILED
N
136
MORE ADDRESS IN QUERY LIST?
Y
138
USE NEXT ADD_FIN IN LIST
N
140
ERROR
142
RELEASE CALL

FIG. 7A

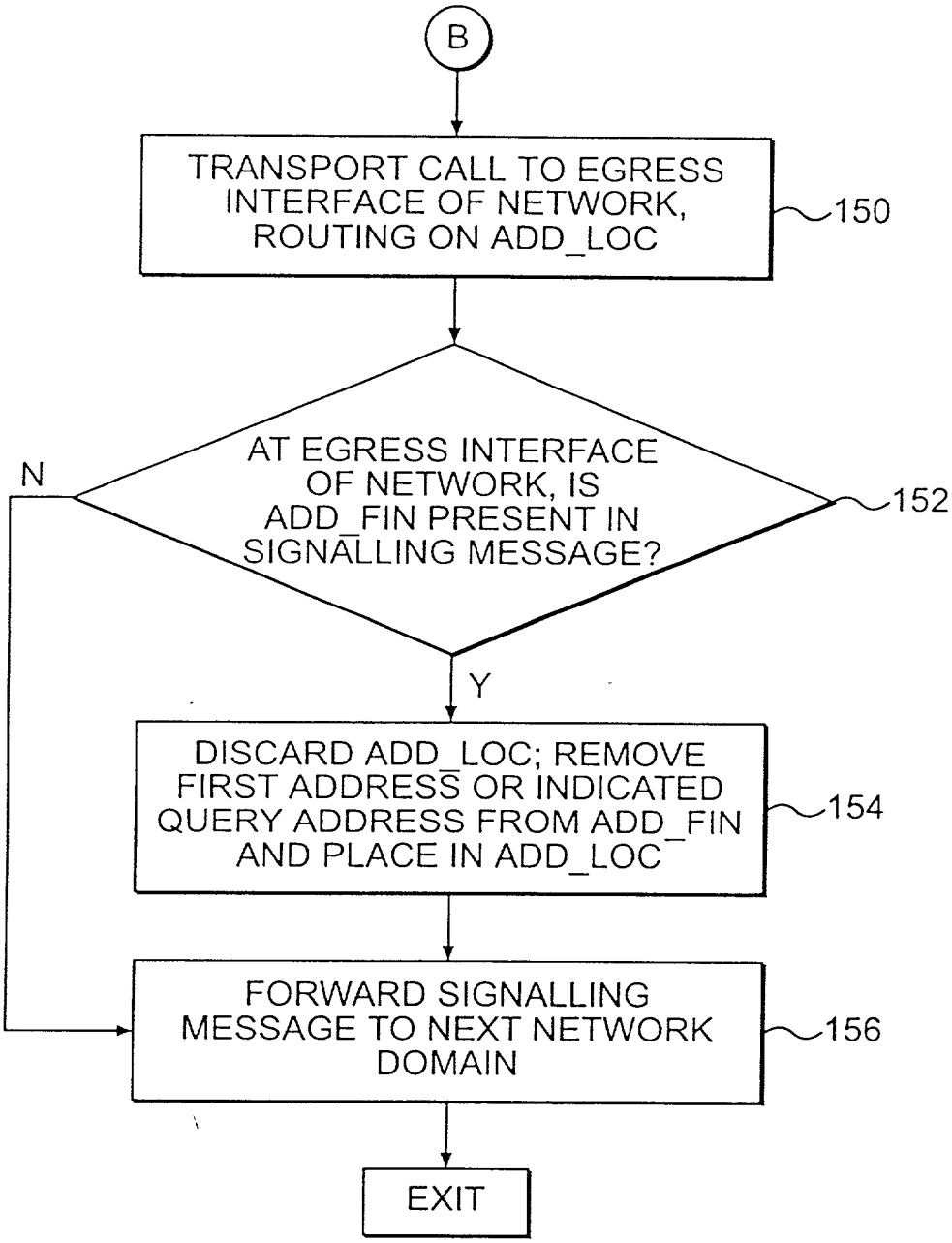

B
TRANSPORT CALL TO EGRESS INTERFACE OF NETWORK, ROUTING ON ADD_LOC
150
AT EGRESS INTERFACE OF NETWORK, IS ADD_FIN PRESENT IN SIGNALLING MESSAGE?
152
N
Y
DISCARD ADD_LOC; REMOVE FIRST ADDRESS OR INDICATED QUERY ADDRESS FROM ADD_FIN AND PLACE IN ADD_LOC
154
FORWARD SIGNALLING MESSAGE TO NEXT NETWORK DOMAIN
156
EXIT

FIG. 7B

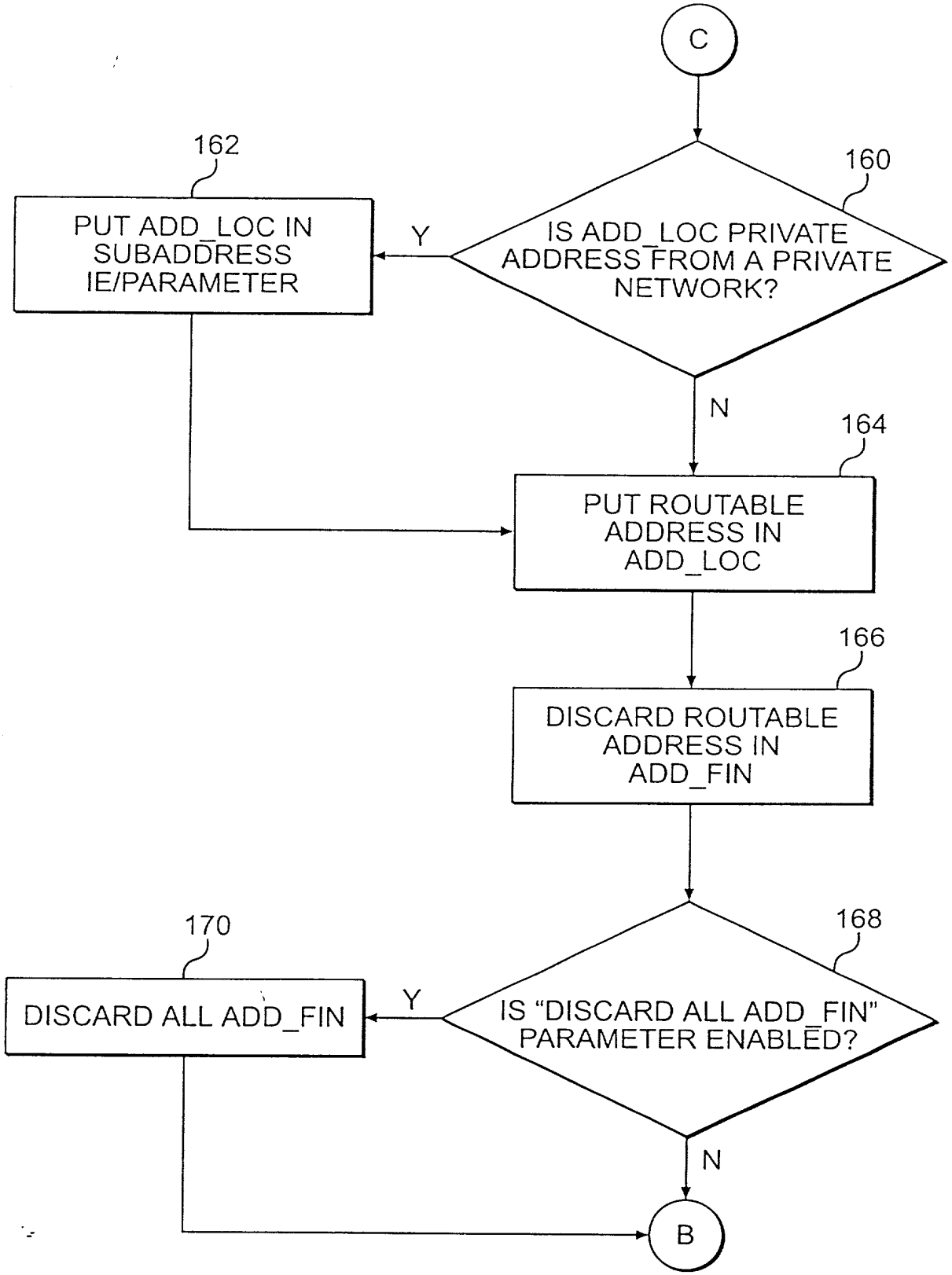

C
160
IS ADD_LOC PRIVATE ADDRESS FROM A PRIVATE NETWORK?
Y
162
PUT ADD_LOC IN SUBADDRESS IE/PARAMETER
N
164
PUT ROUTABLE ADDRESS IN ADD_LOC
166
DISCARD ROUTABLE ADDRESS IN ADD_FIN
168
IS "DISCARD ALL ADD_FIN" PARAMETER ENABLED?
Y
170
DISCARD ALL ADD_FIN
N
B

FIG. 7C

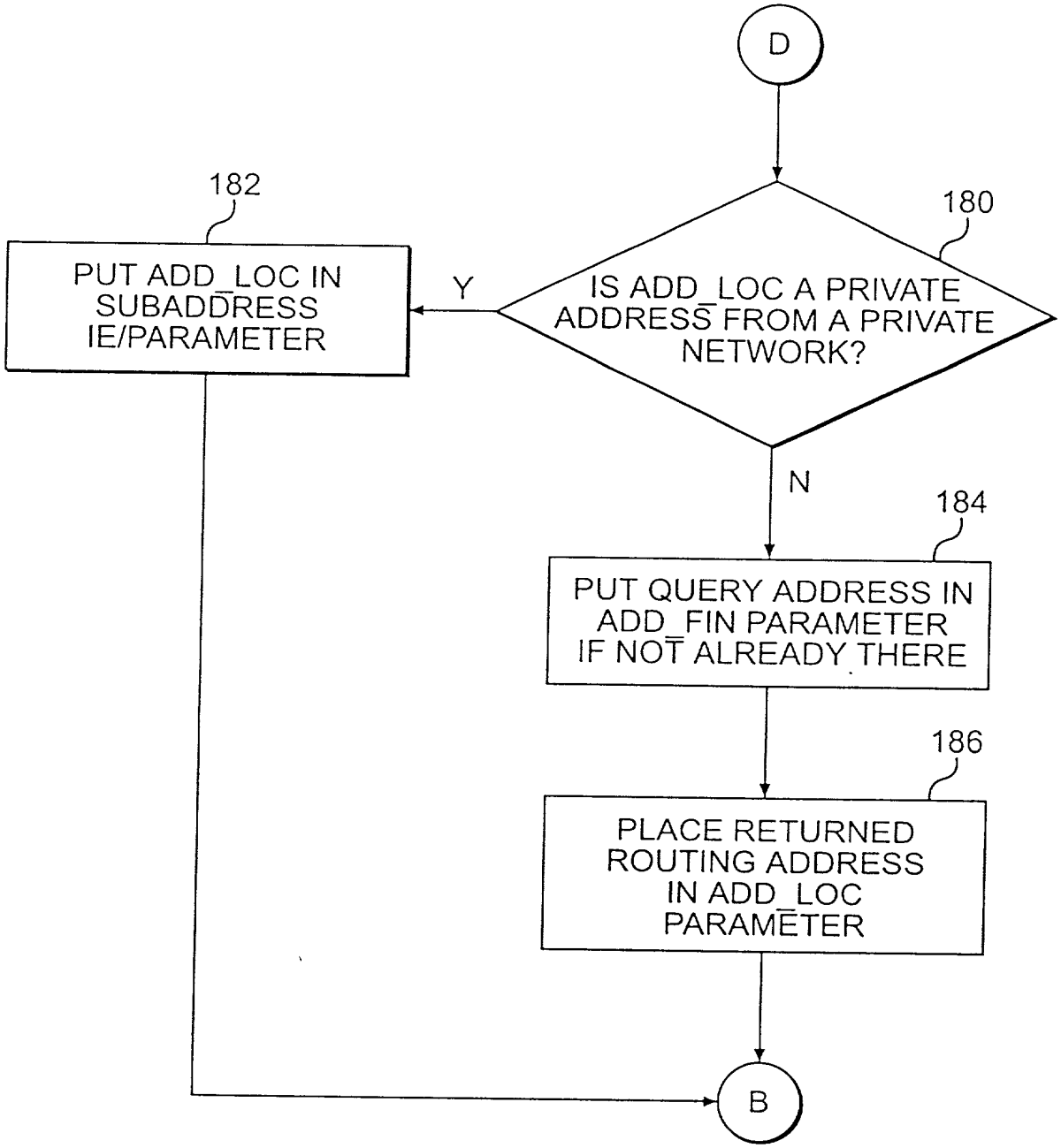

D
180
IS ADD_LOC A PRIVATE ADDRESS FROM A PRIVATE NETWORK?
Y
182
PUT ADD_LOC IN SUBADDRESS IE/PARAMETER
N
184
PUT QUERY ADDRESS IN ADD_FIN PARAMETER IF NOT ALREADY THERE
186
PLACE RETURNED ROUTING ADDRESS IN ADD_LOC PARAMETER
B

FIG. 7D

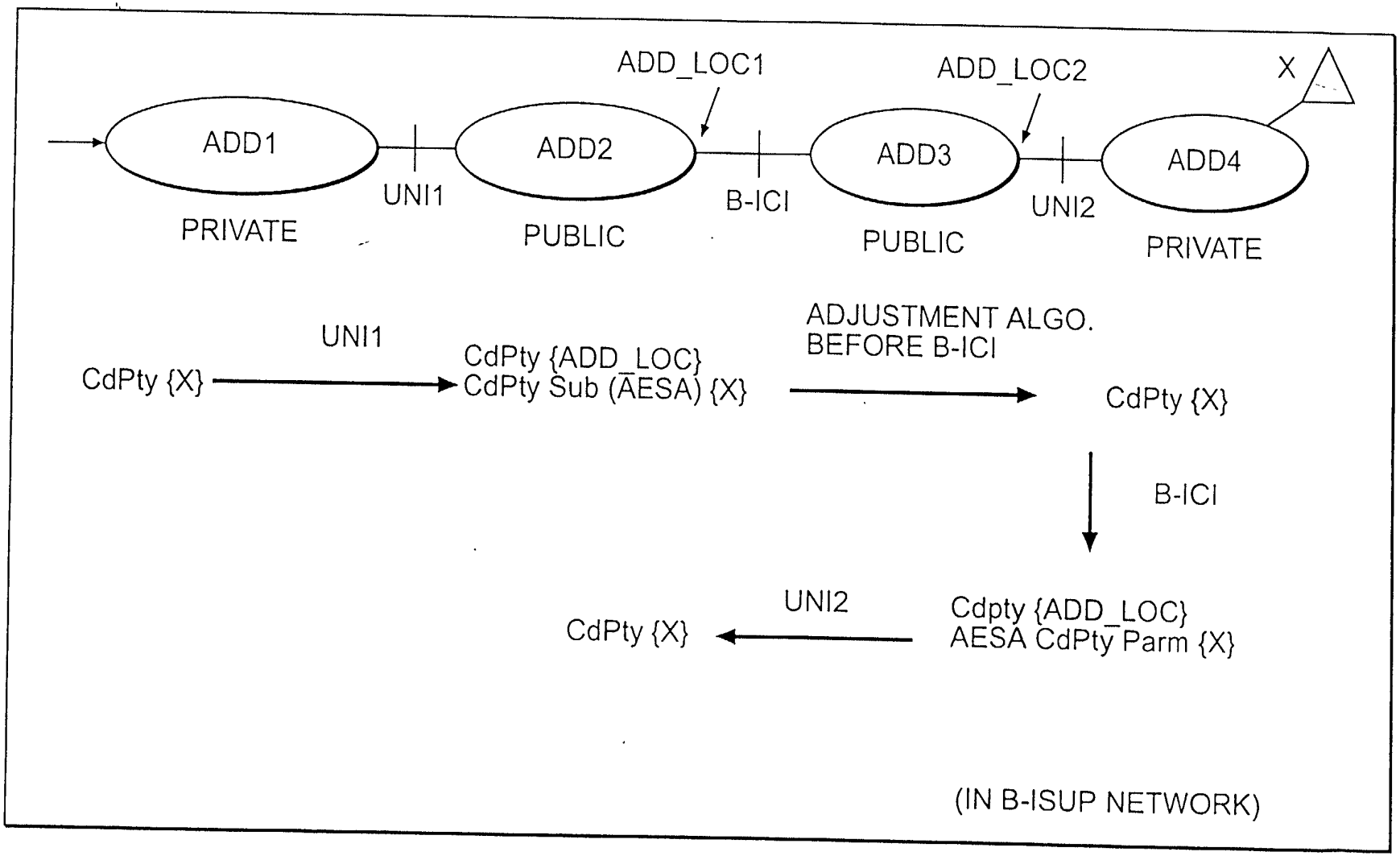

ADD_LOC1
ADD_LOC2
X
ADD1
ADD2
ADD3
ADD4
UNI1
B-ICI
UNI2
PRIVATE
PUBLIC
PUBLIC
PRIVATE
UNI1
CdPty {X}
CdPty {ADD_LOC}
CdPty Sub (AESA) {X}
ADJUSTMENT ALGO.
BEFORE B-ICI
CdPty {X}
B-ICI
UNI2
CdPty {X}
Cdpty {ADD_LOC}
AESA CdPty Parm {X}
(IN B-ISUP NETWORK)

FIG. 8

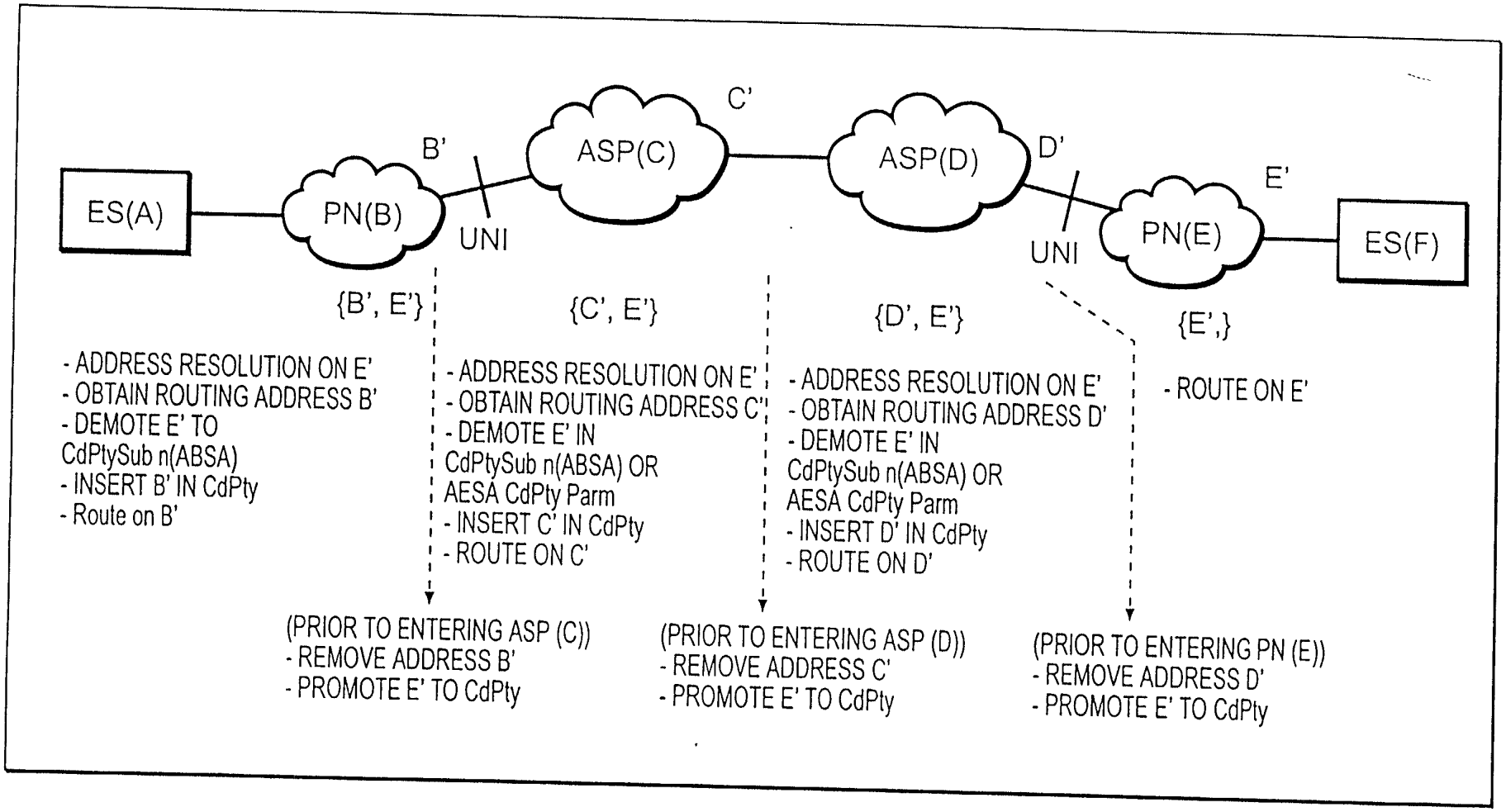


FIG. 9

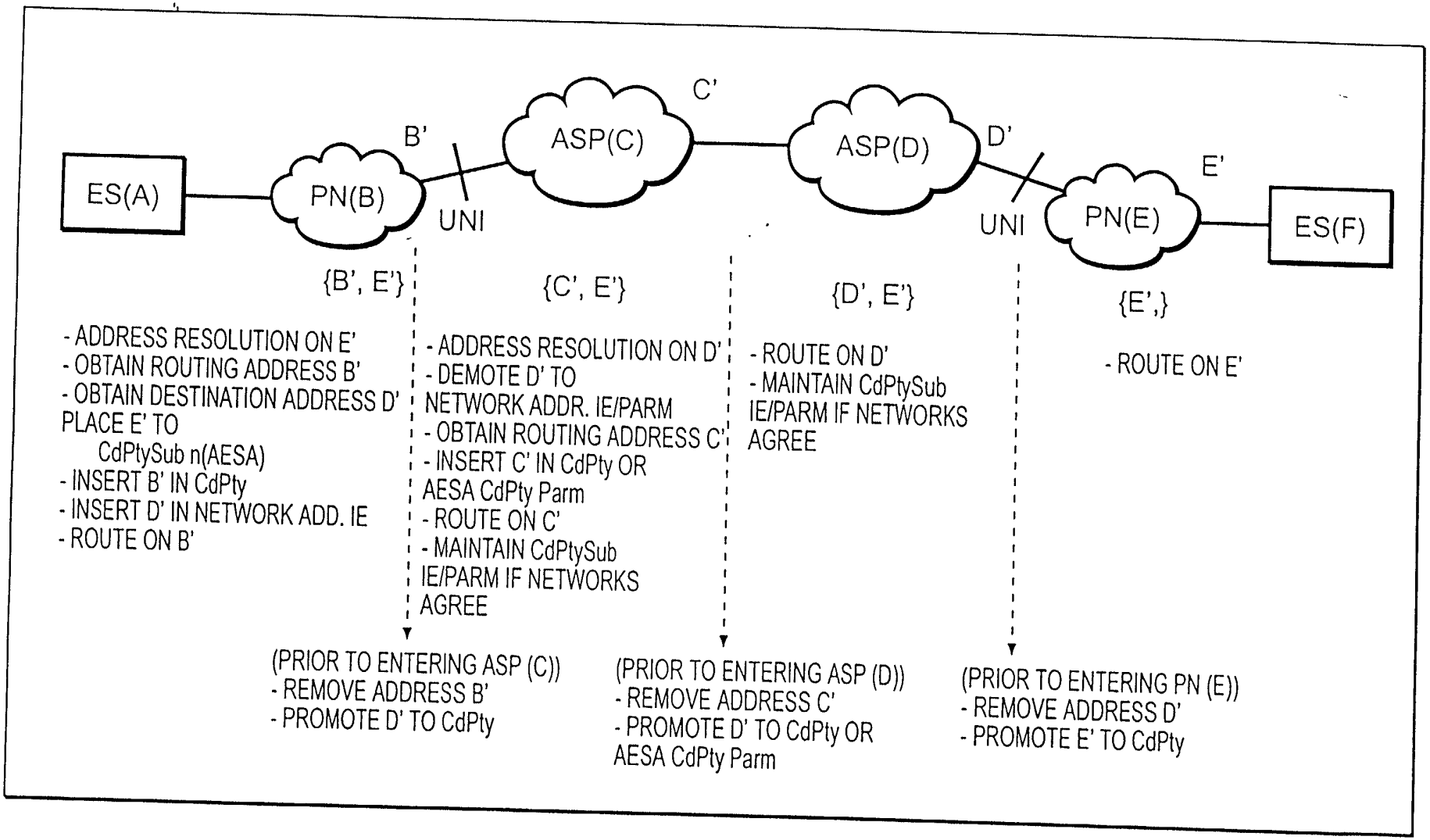
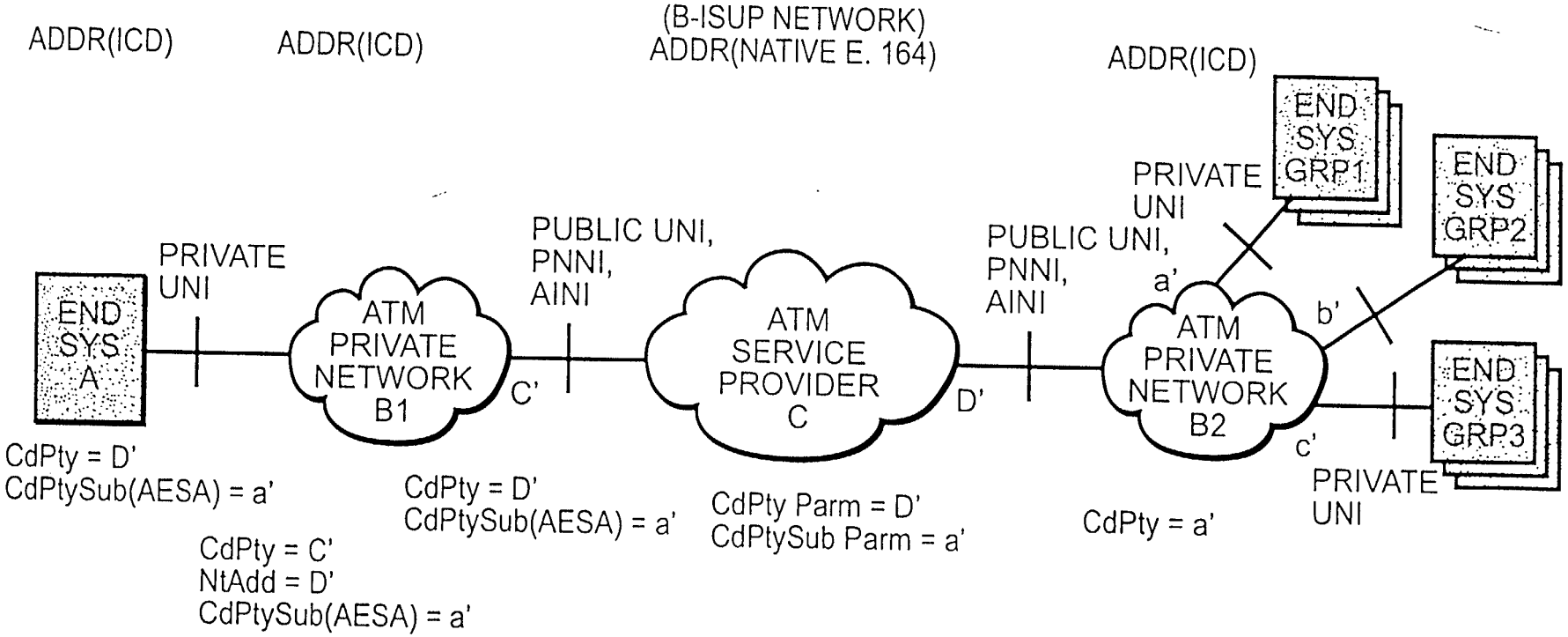


FIG. 10



NOTE: Only address D' is configured in the public network in its routing table or TDB (if it is running OD PNNI). Even when private network mover/changes its address, D' does not need to be changed or reconfigured.

FIG. 11

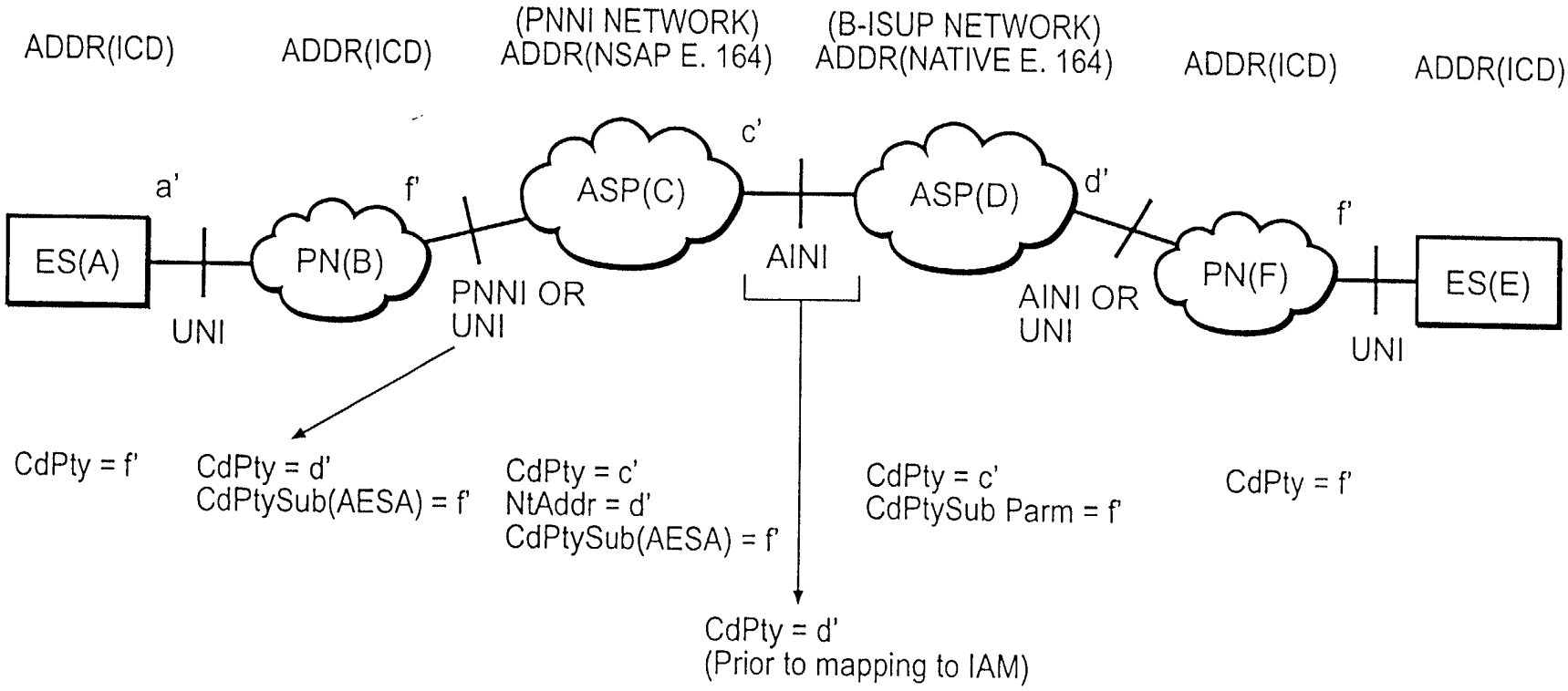
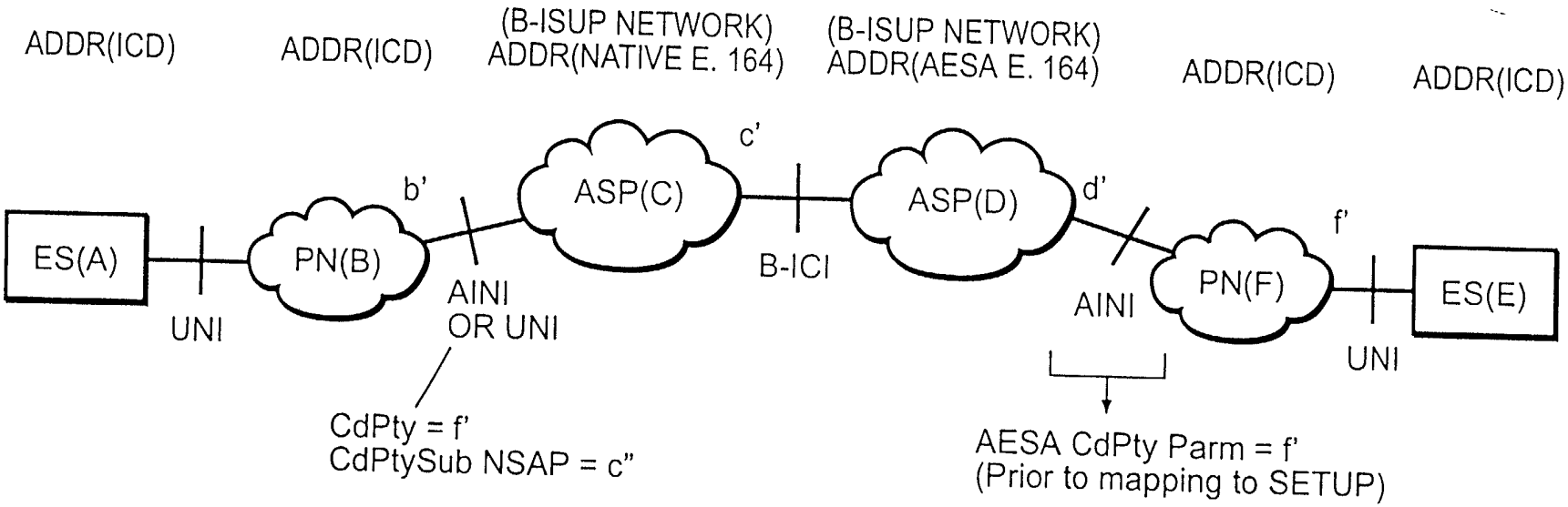
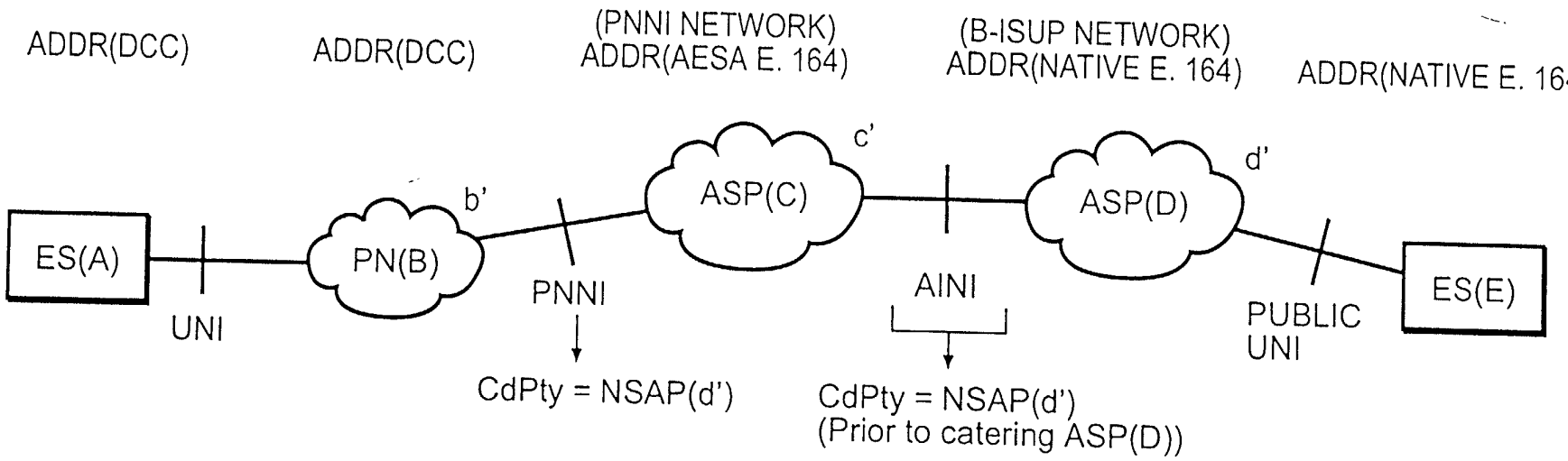

ADDR(ICD)
ADDR(ICD)
(PNNI NETWORK)
ADDR(NSAP E. 164)
(B-ISUP NETWORK)
ADDR(NATIVE E. 164)
ADDR(ICD)
ADDR(ICD)
a'
f'
c'
d'
f'
ES(A)
PN(B)
ASP(C)
AINI
ASP(D)
PN(F)
ES(E)
UNI
PNNI OR
UNI
AINI OR
UNI
UNI
CdPty = f'
CdPty = d'
CdPtySub(AESA) = f'
CdPty = c'
NtAddr = d'
CdPtySub(AESA) = f'
CdPty = c'
CdPtySub Parm = f'
CdPty = f'
CdPty = d'
(Prior to mapping to IAM)

FIG. 12



NOTE: The use of the "CdPtySub(NSAP)" here is to carry the NSAP subaddress which is required by the destination end system ES(E)

FIG. 13



CdPty OR NSAP(d')

CdPty = b'
NtAdd = NSAP(d')

CdPty = c'
NtAddr = NSAP(d')

CdPty Parm = d'
AESA CdPty Parm = NSAP(d')

NOTE: The use of the "CdPtySub AESA" here is to carry the NSAP formal E. 164 AESA with the DSP field act to zero (see UNT 4.0 Spec, guideline #9)

FIG. 14

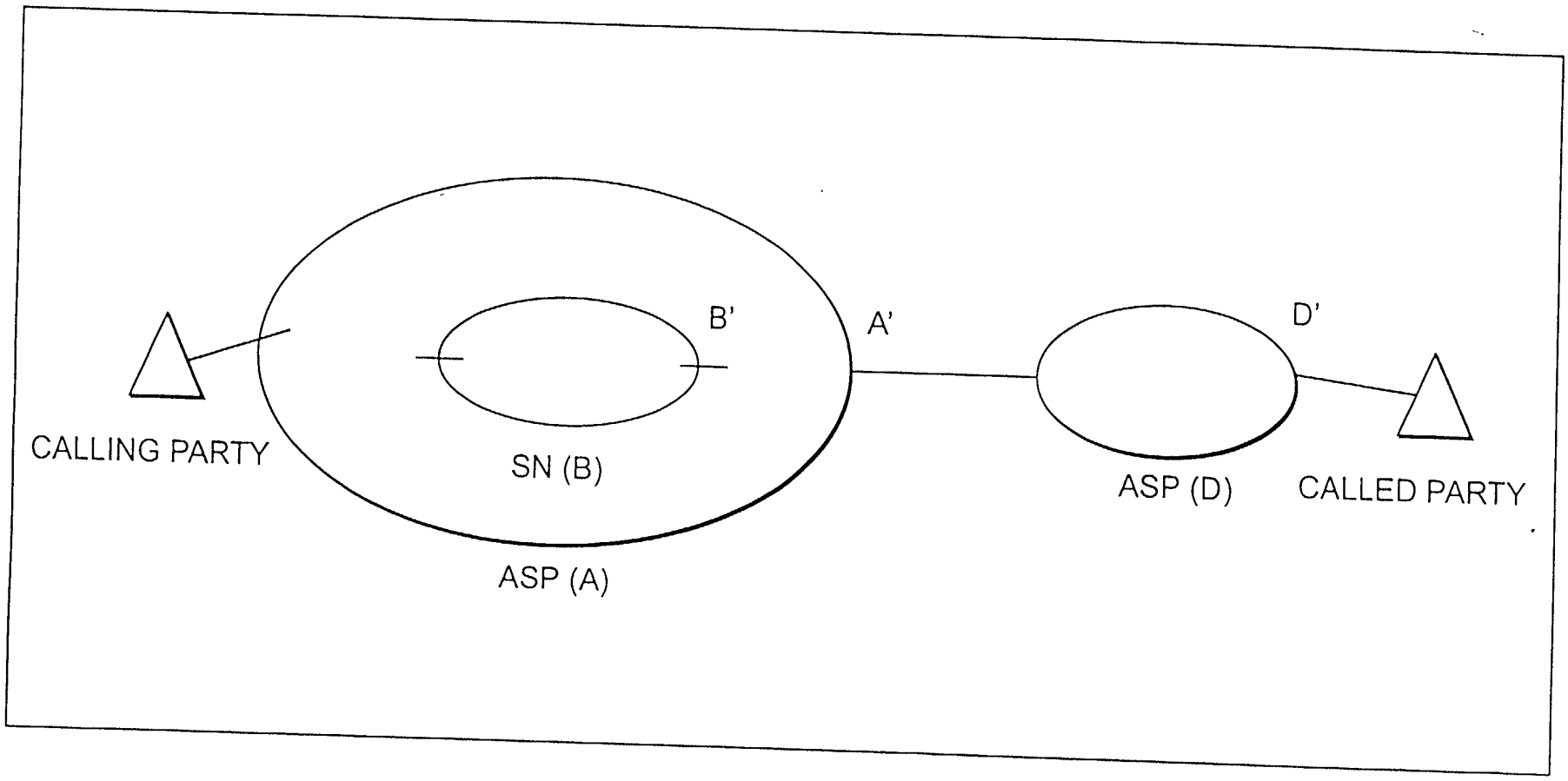


FIG. 15